DOI: http://dx.doi.org/10.26483/ijarcs.v8i7.4572

ISSN No. 0976-5697

Volume 8, No. 7, July – August 2017

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

ASSOCIATION RULE MINING USING BIO INSPIRED BEES SWARM INTELLIGENCE ON CUDA

A. S. Shelar, U. A. Nuli
D.K.T.E Society's Textile and Engineering Institute,
Ichalkaranji,India

Abstract:Association Rule mining (ARM) is well studied and famous optimization problem which finds useful rules from given transactional databases. Many algorithms are already proposed in literature which shows their efficiency when dealing with different sizes of datasets. Unfortunately, their efficiency is not enough for handling large scale datasets. Bio inspiredbees swarm intelligence algorithm for association rule mining is more efficient. These kinds of problems need more powerful processors and are time expensive. For such issues solution can be provided by graphics processing units (GPUs). They are massively multithreaded processors. In this case GPUs can be used to increase performance of the computation. Bees swarm intelligence algorithm for association rule mining can be designed using GPUs in multithreaded environment which will be efficient for given datasets.

Keyword:Bio inspired, Swarm Intelligence, Bees Foraging Behavior, Association Rule Mining, CUDA, GPUs.

1. INTRODUCTION

Association Rule Mining (ARM) is one of the most useful and well known technique of data mining [1]. It is used to extract or retrieve frequent patterns, associations, correlations, or causal structures among sets of items from given datasets. Datasets can be considered as transactional databases, relational databases and other information repositories. Formally, association rule mining problem explained as follows: Let, T be m number of transactions or set of records like {t1, t2, t3,……, tm} from given datasets. And, I be a set of n different items or attributes {i1, i2, i3 ,……..…, in}, an association rule is an implication of the form $X \rightarrow Y$ where $X \subset I$, $Y \subset I$, and $X \cap Y = \emptyset$. The itemset X is called antecedent part (left side) while the itemset Y is called consequent part (right side) and the rule means X implies Y.

Association Rule Mining is related to finding a set of rules considering a large percentage of data and it tends to generate a useful number of rules. However, since the number of transactions are increasingly more, the user no longer looks for all the possible rules but user looks to determine only a subset of important rules. To measure usefulness of association rules, mainly two basic parameters are used, one is namely support of a rule and second is confidence of rule. The support of an itemset $I \subseteq I$ is the number of records containing I. The support of a rule $X \rightarrow Y$ is the support of $X \cup Y$ and the confidence of a rule is support $(X \cup Y)$ / support (X). An association rule $X \rightarrow Y$ with a confidence of 70% means that 70% of the transactions that contain X also contain Y together. So, the association rule mining is to find important rules which are having support$\geq$ MinSup and confidence $\geq$ MinConf [2]. Here, MinSup and MinConf are two thresholds predefined by users.

Swarm intelligence is based on the collective behaviour of decentralized, self-organized systems. Many researchers are interested in this new existing way of achieving a form of artificial intelligence including simple agent groups. Modelling the behaviour of social insects such as ants, bees, firefly, bat etc. and using these models for search and problem-solving are the context of the emerging area of swarm intelligence. Bees are among the well-studied social insects [3]. Many researchers studied the bees behaviour in order to design powerful methods.

The bees behaviour is divided into three categories: marriage behaviour, foraging behaviour and the queen bee. Marriage bees behavior [4] approach, is used in honey bees optimization. Bees are social insects living in organized colonies. Each honey-bees colony consists of one or several queens, drones, workers and broods. Queens specialize in egg laying, workers in brood care and sometimes egg lying, drones are the males of the colony and broods the children.

The meta-heuristic BSO proposed in [5] is inspired by the foraging bees behaviour. It is based on a swarm of artificial bees cooperating together to solve a problem. The general functioning of this is as follows: First, a bee named Initial Bee settles to find a solution presenting good features. From this first solution called Sref we determine a set of other solutions of the search space by using a certain strategy. This set of solutions is called Search Area. Then, every bee will consider a solution from Search Area as its starting point in the search. After accomplishing its search, every bee communicates the best visited solution to all its neighbors through a table named dance. One of the best solutions stored in this table will become the new reference solution during the next iteration. In order to avoid cycles, the reference solution is stored every time in a taboo list. The reference solution is chosen according to the quality criterion. However, if after a period of time the swarm observes that the solution is not improved, it introduces a criterion of diversification preventing it from being trapped in a local optimum. The diversification criterion consists to select among the solutions stored in taboo list, the most distant one. The algorithm stops when optimal solution is found.

Nowadays Graphics processing units (GPUs) is fast and cheap parallel hardware traditionally used to speed up 3D

graphic applications. GPU-accelerated computing is the use of a graphics processing unit (GPU) together with a CPU to accelerate deep learning, data analytics, and engineering applications. It is pioneered in 2007 by NVIDIA, GPU accelerators now power energy-efficient data centers in government labs, universities, enterprises, and small-and-medium businesses around the world. They play a huge role in accelerating applications in platforms ranging from artificial intelligence to cars, drones, and robots. This new technology general purpose graphics processing units are also known as GPGPU. GPU-accelerated computing offloads compute-intensive portions of the application to the GPU, while the remainder of the code still runs on the CPU. From a user's point of view, applications simply run much faster. Hence it used to improve execution time of various computation from different domains.

Motivating by graphics processing units (GPUs), the power of GPUs can be used to solve highly computational problems from computer science domain. So, a bees swarm optimization for association rule mining using GPUs is useful approach than serial approach of Bees swarm Intelligence. The evaluation process of the solutions could be done on GPU because of GPUs massively multithreaded environment.

2. RELATED WORK

Some well-known algorithms have been proposed for generating association rules are AIS [6], Apriori [7] and FP-Growth [8].

Agrawal R, I mielinski T and Swami A,[6] have proposed AIS algorithm which is very space consuming and requires too many passes over the whole database. Agrawal, R. and Ramakrishan, S [7] has given Apriori algorithm which is the best well-known algorithm for association rules mining. It is basically based on breadth first search (BFS) strategy to count the supports of itemset and uses a candidate generation function to achieve the downward closure property of support. Han, J., Pei, J., Yin, J., Mai, R. [8], have been proposed FP-growth algorithm. It uses an FP-tree construction to wrap the database and a divide-and-conquer approach, to decompose the mining tasks and the database as well. But tree generation is always complex part of data organization.

Agrawal and Shafer [9] projected two parallel versions of Apriori called count distribution (CD) and data distribution (DD). These are leading parallel ARM algorithms. In CD the dataset is divided in between several processors and each processor executes the entire Apriori on its part of data.Beyond all these algorithms, different parallel metaheuristics have been proposed in the literature for ARM problem. Melab N, Talbi E-G [10] proposed a parallel genetic algorithm (GA), called PGARM, based on the master/workers model.

Following this brief state of the art, different ways have been proposed for parallel ARM problem. For each approach, the authors take advantage of the used parallel hardware. However, all these algorithms have been implemented on old-fashioned parallel architectures which are still expensive and not always available for everyone.

Since 2007, GPU technology, fast, cheap, and parallel hardware, usually available on most computers, has been successfully used on many domains. For this, ARM community is also investigating on this simple but reliable technology.To the best of our knowledge, the only work which introduces metaheuristics for ARM on GPU is proposed in [11] by Cano et al. In this paper, an evolutionary algorithm is proposed to solve the ARM problem on GPUs.

Djenouri Y, Drias H, Habbas Z in [12], [13] applied Bees algorithm for web association rule mining and bees swarm optimization using multiple strategies for association rule mining respectively. In [14] recent Youcef Djenouri, Habbas proposed algorithm for GPU based bees swarm optimization for association rule mining.

3. METHODOLOGY

3.1 ParallelApproach: Bees Swarm Intelligence for Association Rule Mining on CUDA

The proposed system is designed and developed with following modules, Fig 3.1 which are given as below:

1. Create Initial Solution
2. Search Area Determination
3. Neighborhood Computation
4. Fitness calculation
5. Dance Table

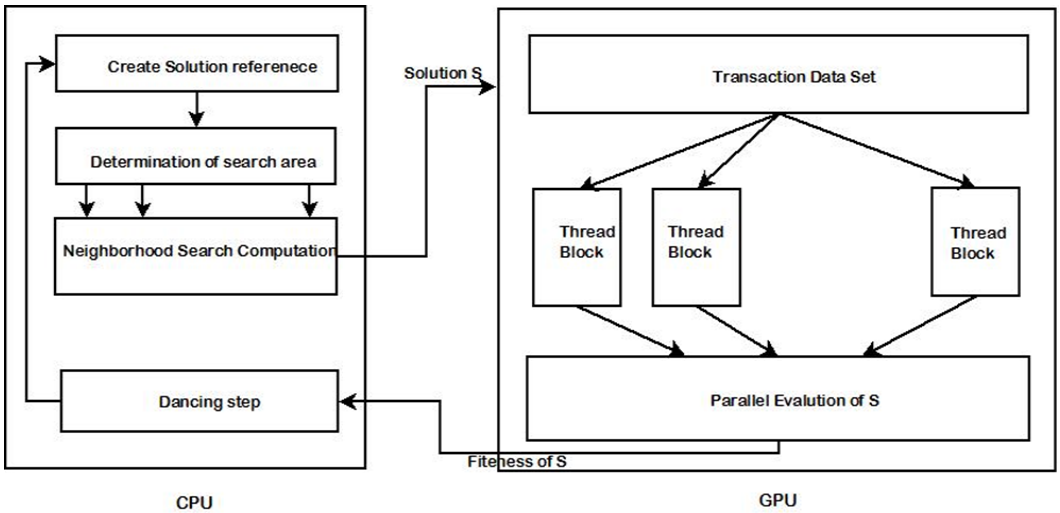

Fig. 3.1Bees Swarm Intelligence for Association Rule Mining on CUDA

The mentioned module details are given as follows.

3.1.1. Create Initial Solution

Generate initial solution randomly for considering N items. Solution representation is important consideration in this project. Integer encoding representation allows to separate the antecedent part and the consequent part of the association rule. And, calculate cost of the initial solution which is based on support and confidence of the given solution. The cost is known in terms of system is fitness for given solution.Rule is considered as one solution in the search space, each one is represented by a vector S of N bits and their positions are defined as follows:

- S[i] = 0 if the item i is not in the solution S.
- S[i] = 1 if the item i belongs to the antecedent part of the solution S.
- S[i] = 2 if the item i belongs to the consequent part of the solution S.

Example:

Let T: {t1, t2, ……. , t5} be a set of items
S1: {1, 1, 0, 0, 2} represents the rule
R1: t1, t2 ⇒ t5.

Below Fig. 3.2 shows detail about how to represent rule. (Integer Representation). For given rule only three items which are considered Bread, Peanuts and Jam. Bread and Peanuts are antecedent part of the rule and Jam is consequent part of the rule.

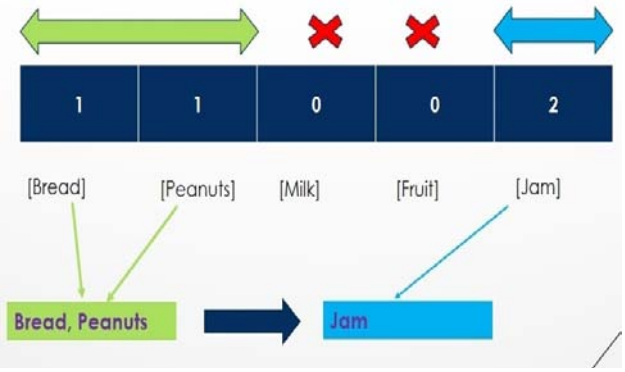

Fig. 3.2 Integer Encoding Rule Representation

3.1.2 Search Area Determination

Given an Initial reference solution Sref and a colony of K bees the search area operations determines K search spaces, each one is associated to a bee.

Each bee k builds its own search area by changing successively in the solution Sref the bits k + i × Flip where i varies from 0 to n− 1 and Flip is a given parameter. This strategy can be used if and only if the number of bees is less or equal to N/Flip.

For the search area, the aim is to determine the regions of the bees using Sref already created. The strategies have been developed to explore search area. For example, strategy aims to perform flip jump on Sref. If we have k=3 and flip=2 and N=5,

We obtain:

- The first bee is obtained by modifying the bits (1, 3, 5)
- The second bee is obtained by modifying the bits (2, 4) and
- The last bee is obtained by modifying the bits(3, 5)

3.3.3 Neighborhood Computation

The neighborhood computation for each search area is obtained by changing from a given solution S one bit in a random way. Based on this simple operation, N neighborhoods are created

Example:

Consider the given solution: S = {1, 0, 0, 1, 2}

1. Change the first bit in S: S1 = {0, 0, 0, 1, 2}
2. Change the second bit in S: S2 = {1, 2, 0, 1, 2}
3. Change the third bit in S: S3 = {1, 0, 1, 1, 2}
4. Change the fourth bit S: S4 = {1, 0, 0, 0, 2}
5. Change the fifth bit S: S5 = {1, 0, 0, 1, 0}

All neighbors send serially to evaluate fitness of the solution.

3.3.4 Fitness

In this Module for each generated solution (rule) from neighborhood computation, the entire transactional database is scanned. The solution fitness is based on the support and the confidence of the given rule which is computed as follows:

$$\text{Fitness}(s) = \alpha \times \text{confidence}(s) + \beta \times \text{support}(s) \qquad (3.1)$$

This function should be maximized. For each invalid solution s where Sup(s) < Minsup or Conf(s) < MinConf, the Fitness(s) is set to −1 and the solution is rejected.

3.3.5. Dance Table

Each bee puts in the dance table the best rule found among its search. The communication between bees is done in order to find the best dance (the best rule) which becomes the reference solution for the next pass. The general functioning of the algorithm is as follows: First, the solution initial reference (Sref) is initialized arbitrarily so that each element of Sref belongs to {0, 1, 2}. After that, excluding the Fitness Computing which is applied for each generated solution, the other steps are repeated in the order until maximum iteration is reached.

These five modules combinally work on using CPU and GPU. Fig.3.1 shows working of these modules. CPU runs master model and GPU runs slave which consists of evaluation fitness for given solution.

4. HELPFUL HINTS

Implementation of Parallel approach of proposed system is carried out using CUDA enabled GPUs. CUDA is a parallel computing platform and application programming interface (API) model created by NVidia. It permits software developers and software engineers to use a CUDA-enabled graphics processing unit (GPU) for general purpose processing – an approach termed GPGPU (General-Purpose computing on Graphics Processing Units). The CUDA platform is a software layer that gives uninterrupted access to the GPU's instruction set and parallel computational elements.

Above proposed system is design and developed using general approach of bees swarm intelligence algorithm. Algorithm 4.1 gives general bees algorithm.

Algorithm 4.1: The general Bee Swarm Intelligence algorithm

Input: transactional database

Output:number of solutions

1. Sref ← The solution found by Initial Bee
2. while i < Max_Iteration and not stop do
3. Insert Sref in taboo list.
4. Search-Area(Sref)
5. Assign a solution from SearchArea to each bee
6. for each bee K do
7. Built-Search-Area (beek)
8. Store the result in the table Dance
9. Communication between bees to choose best solution
10. end for
11. Choose the new reference solution Sref
12. end while

5. EXPERIMENTAL SETUP AND RESULT

The proposed system is performed using specific software and hardware requirements.

5.1 Software requirements for experiment:

Proposed system is implemented using Windows 10, 64-bit Operating System, using C as programming language. Visual Studio 12.0 with CUDA 7.0 Runtime Environment**.**

5.2 Hardware requirements for experiment:

Intel(R)Core(TM) i5–4440Processor with3.10 GHz clock speed, 4 GB memory and machine having NVidia graphics card GT 730 DDR3 64-bit 2GB memory with 386 cores.
Proposed System is tested using real time dataset with named Skin. Skin dataset is originally having 11 distinct items and total 245057 records. Below Fig. 5.1showsperformance difference between serial vs. parallel BSI-ARM system. Here, performance is analyzed considering different parameters like K (search area),Flip and IMAX. Where Flip=4, IMAX=100 and K is considered for different values K=5, K=7, K=9.

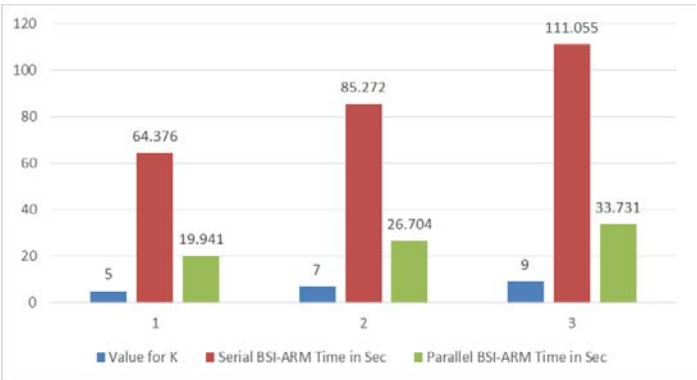

Fig 5.1 Serial vs. Parallel BSI-ARM

6. CONCLUSION

Bio inspired swarm intelligence based approaches are more powerful than traditional problem solving techniques. If these approaches are applied to solve highly computational problems like data mining, analysis problems on large scale dataset, then it is time consuming task. In that case GPGPU can be used to divide this task and solve in parallel using massively multithreaded environment of GPU. In this system so many alternatives still possible to map the association rule mining problem to GPU processors. Hence, in future it has remarkable chances to improve the GPU processing using proper allocation of problem with great memory utilization and less time consumption.

REFERENCES

[1] Han, J., Kamber, J. and Pei, M. (2011) Data Mining: Concepts and Techniques, Vol. 8, 3rd ed., pp.1–50, China Machine Press.

[2] Agrawal, R. and Shafer, J. (1996) 'Parallel mining of associations rules', IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 6, pp.962–969.

[3] Bessedik, M., Bouakline, T. and Drias, H. (2011) 'How can bees colour graphs', Int. J. Bio-Inspired Computation, Vol. 3, No. 1, pp.67–76.

[4] Pham, D.T., Castellani, M. (2009), The Bees Algorithm – Modelling Foraging Behaviour to Solve Continuous Optimisation Problems. Proc. ImechE, Part C, 223(12), 2919-2938.

[5] Drias, H., Sadeg, S. and Yahi, S. (2005) 'Cooperative bees swarm for solving the maximum weighted satisfiability problem', in Proceedings of IWANN, pp.318–325.

[6] Agrawal R, I mielinski T and Swami A, Mining association rules between sets of items in large databases, Proceedings of the ACM SIG2 MOD,Washington DC, pp 207- 216, 1993.

[7] Agrawal, R. and Ramakrishan,S.: Fast algorithms for association rules in large databases (http://rakesh.agrawalfamily.com/papers/vldb94apriori.pdf) , in Bocca, Jorge B.; Jarke, Matthias; and Zaniolo, Cqarlo; editors, Proc of the 20th International Conference on very large Data bases -VLDB), Santiago, Chile, PP 487-499, Sept 2004.

[8] Han, J., Pei, J., Yin, J., Mai, R.: Mining frequent patterns without candidate generation, in Data Knowledge and Knowledge discovery, No 8, PP 53-87, 2004.

[9] Agrawal R, Shafer JC (1996) Parallel mining of association rules. IEEE Trans Knowl Data Eng 8(6):962–969.

[10] Melab N, Talbi E-G (2001) A parallel genetic algorithm for rule mining. In: Proceedings of the 15th international parallel and distributed processing symposium. IEEE Computer Society, London

[11] Cano A, Luna JM, Ventura S (2013) High performance evaluation of evolutionary-mined association rules on GPUs. J Supercomput 1–24

[12] Djenouri, Y., Drias, H., Habbas, Z., Mosteghanemi, H.: Bees Swarm Optimization for Web Association Rule Mining. In: 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT), vol. 3, pp. 142–146. IEEE (2012)

[13] Djenouri, Y., Drias, H., Chemchem, A.: A hybrid Bees Swarm Optimization and Tabu Search algorithm for Association rule mining. In: 2013 World Congress on Nature and Biologically Inspired Computing (NaBIC). IEEE (2013)

[14] Djenouri, Y., Bendjoudi A, Mehdi M, Nadia N, Habbas Z :GPU-based bees swarm optimization for association rule mining in: J Supercomput(2015) 71:1318-1344